Model-based Measurements Of Network Loss

June 28, 2013

Mirja Kühlewind
mirja.kuehlewind@ikr.uni-stuttgart.de

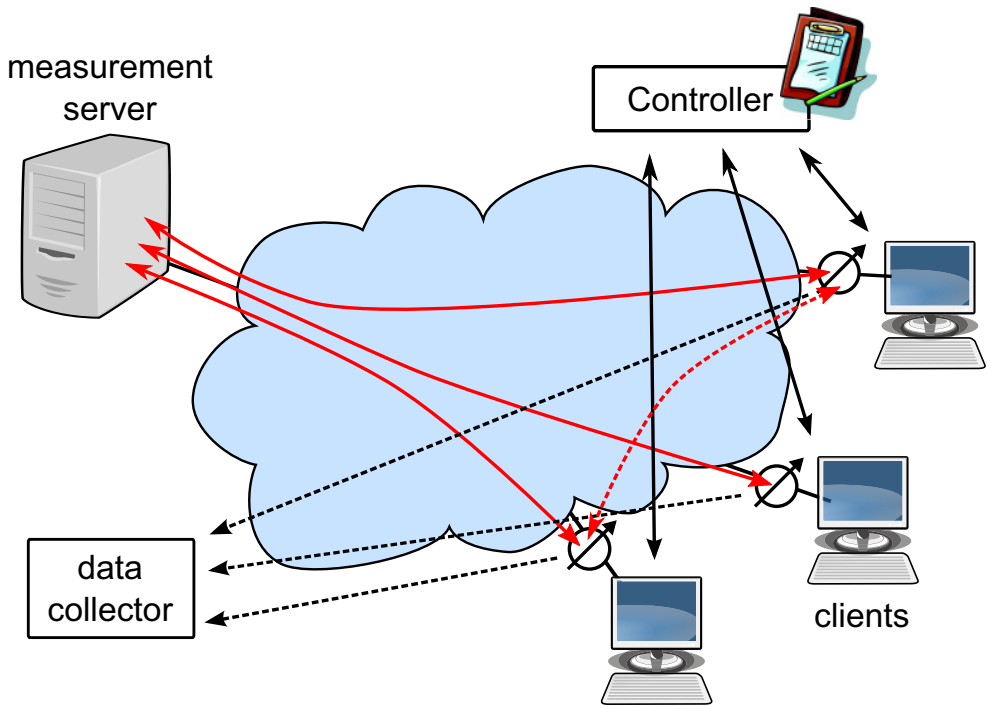
Universität Stuttgart
Institute of Communication Networks
and Computer Engineering (IKR)
Prof. Dr.-Ing. Andreas Kirstädter

LMAP – Architecture Overview

Large Scale Measurement of Access Network Performance

→ Started activity in the Internet Engineering Task Force (IETF) due to request of regulators to evaluate the state of broadband performance

see draft-schulzrinne-lmap-requirements-00



- **Measurement Client** performs active measurements against a measurement server
- **Measurement controller** schedules measurements
- **Data collector** is the reference point to send measurements results for data evaluation

Model-based Measurements

→ Model-Based Internet Performance Metrics

see draft-mathis-ippm-model-based-metrics-01

Question (User / Content provider view)

Is a certain (class of) application(s) usable over a given end-2-end path?

Approach

Active measurement before and during service usage

- Determine (sub)path characteristics
- Calculate performance target value based on path characteristics and application requirements
- Check if a given end-to-end path meets predefined application performance targets
- (Optional) If not, figure out where the problem is (content provider access, user access, network/peering point)

Loss Measurements

Outline

Introduction

Modeling of Loss Pattern
- Simulation setup and scenario
- Traffic Model
- Loss Pattern

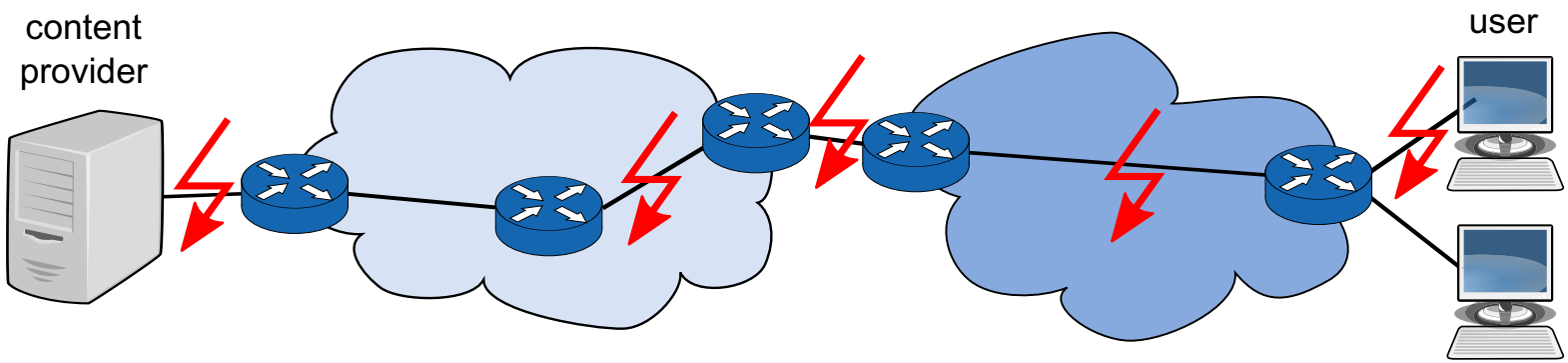
Burst Loss Measurements
in real Internet usage scenarios

Conclusion

Loss Measurements

Goal

Identifying the conditions and potentially origin of losses / congestion



Sources of loss

- Network
  - Transmission errors
  - Overload (shared congestion)

  → Spatial and temporal correlation of loss measurements needed
- Host behavior

  Probing/Start-up of TCP congestion control (self-congestion on access link)

  → Analysis of loss pattern in typical Internet usage scenarios needed

Modeling of Loss Pattern

Goal

Create scenarios with border case loss patterns (e.g. large, periodic loss bursts)

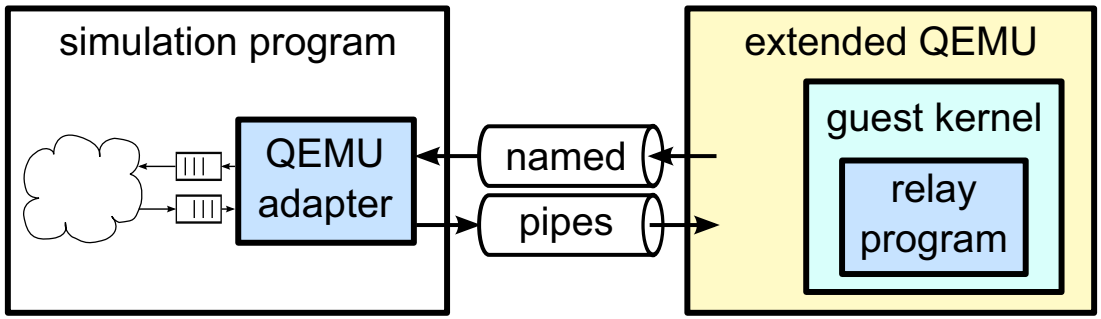
(because application performance does not depend on average loss rate but the average loss rate is strongly influenced by application behavior and congestion control)

Method

- Event-driven simulation using real kernel code
  - → Generate realistic TCP traffic behavior
  - → Controlled loss-free network environment
- Traffic Model
  - Generate flows with certain data size and inter-arrival time (IAT)
  - Different flow sizes and IATs to exploit loss patterns induced by congestion control algorithm (TCP cubic)
- Evaluation based on burst loss metric (number of losses within one RTT after first loss)

  captures loss pattern in a congestion control-aware way (probing and reaction time of congestion control determines periodic loss events)

Event-driven Simulation using Real Kernel Code



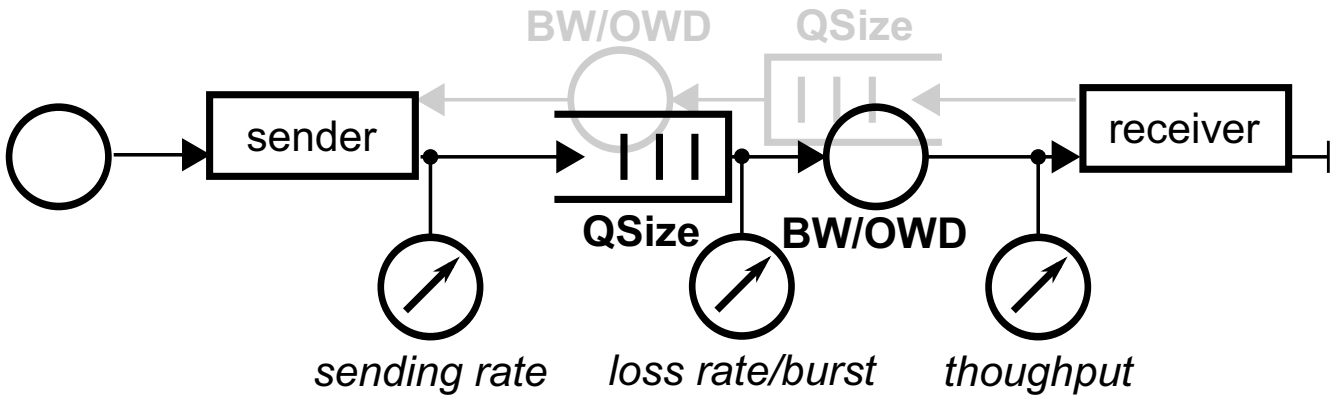
Integration of virtual machines (using QEMU) in the IKR SimLib

- Java-based, event-driven simulation framework (IKR SimLib)
- Kernel code integration by using a Linux OS in a virtual machine (QEMU)
- Control of timing and interfaces for event handling in the simulation program
- Relay program in the guest OS to generate TCP traffic
- Real TCP/IP packets with dummy data are forwarded over simulated network

→ Simulation framework with realistic TCP behavior

→ Easy updates to latest kernel versions possible

Simple Scenario



Network Parameter

- One-Way Delay (OWD) = 50 ms
- Bandwidth (BW) = 5.7 Mbit/s
- Queue size = 71250 Byte (Bandwidth-Delay-Product) = 47.5 full packets

Traffic Model

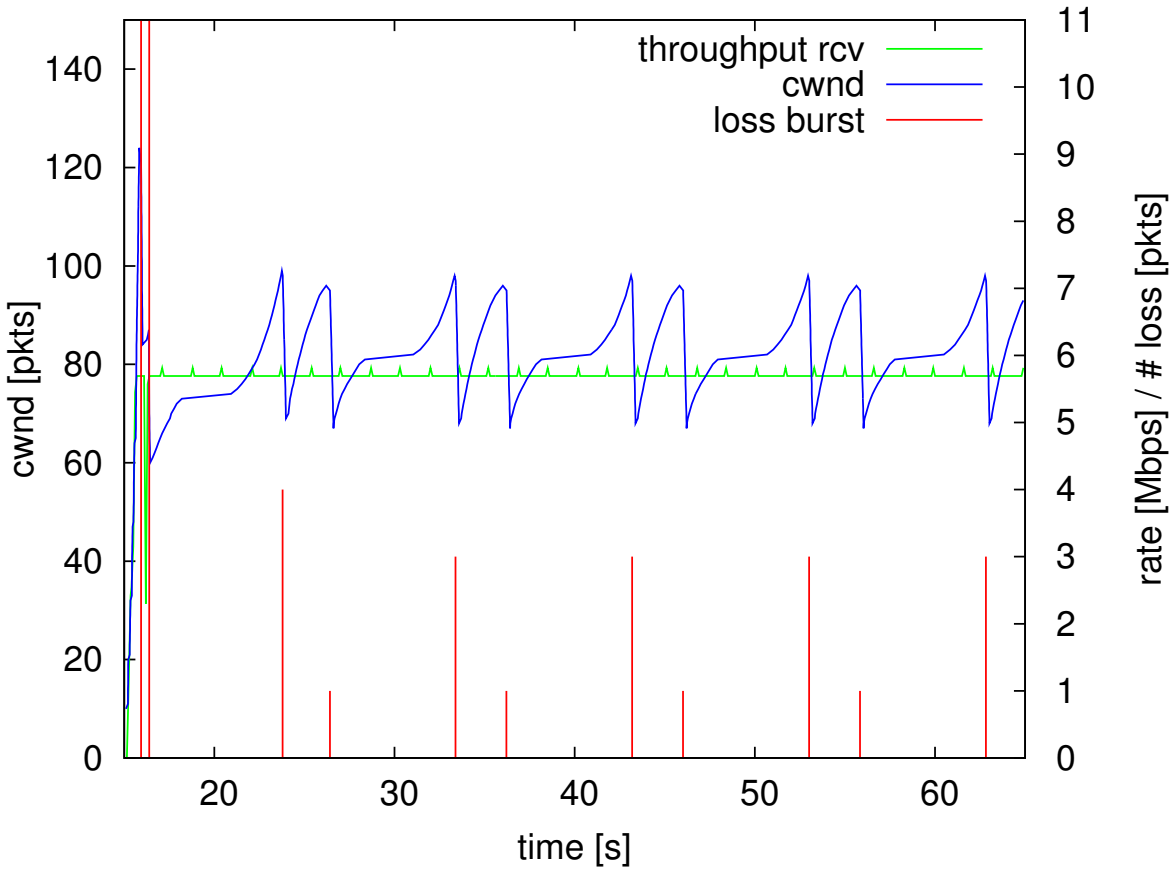
Traffic Generation

- Greedy traffic
  - Data to send are always available
  - Transmission starts at the beginning of simulation
- Regular bursts
  - Fixed size bursts to fill the link and enlarge congestion window to maximum size
  - Only short idle time between bursts to keep the congestion window large

  → Bursts of packets are sent out at once at the beginning of each transmission as no packets are in flight but the congestion window is large
- Random short flows
  - Negative exponential inter-arrival time (IAT)
  - Uniformly distributed small number of packets per flow

  → Cause Slow Start overshoot (80 packets needed to fill the empty link with an initial congestion window of 10 packets)

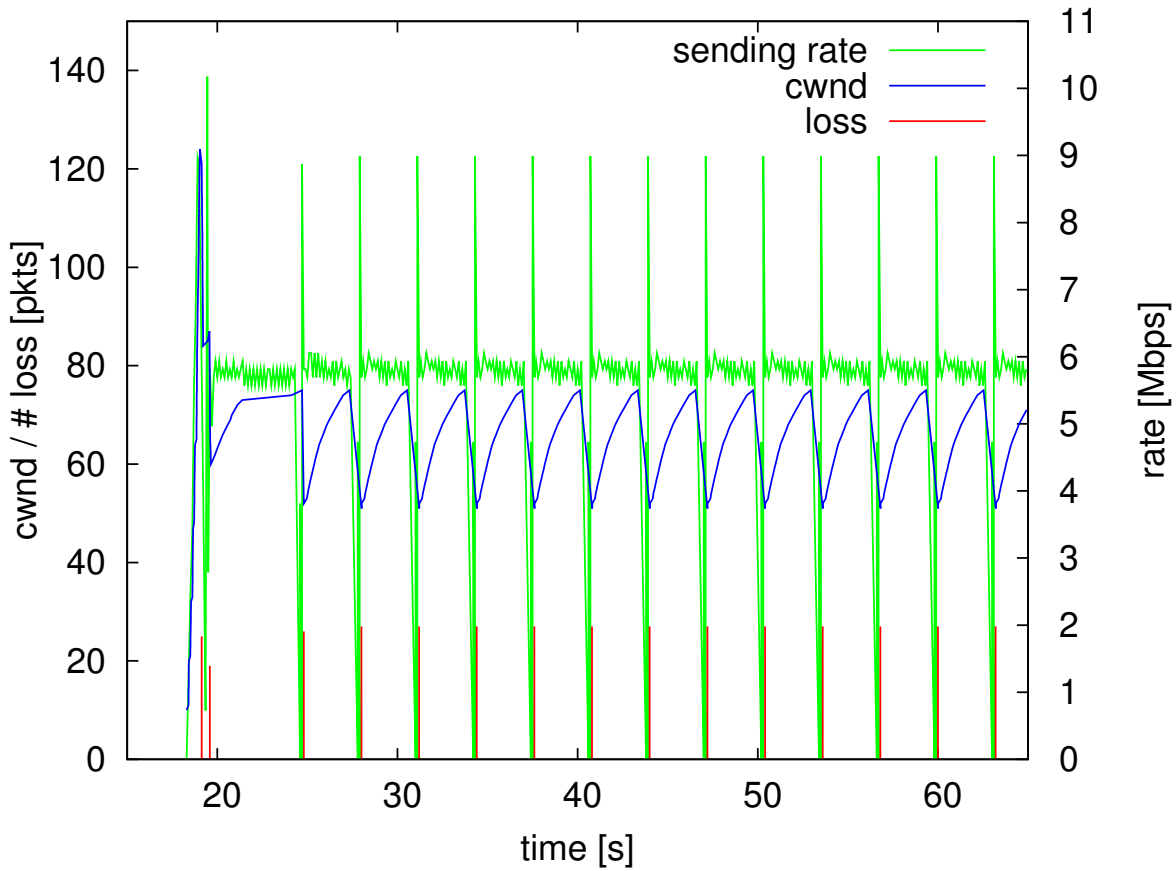
Regular Loss Pattern with Greedy Traffic



→ Regular loss pattern with periodic loss distance (1 or 3 packets per burst loss and a loss distance of 2.82 sec or 6.98 sec)

Regular Loss Pattern with Bursty Traffic

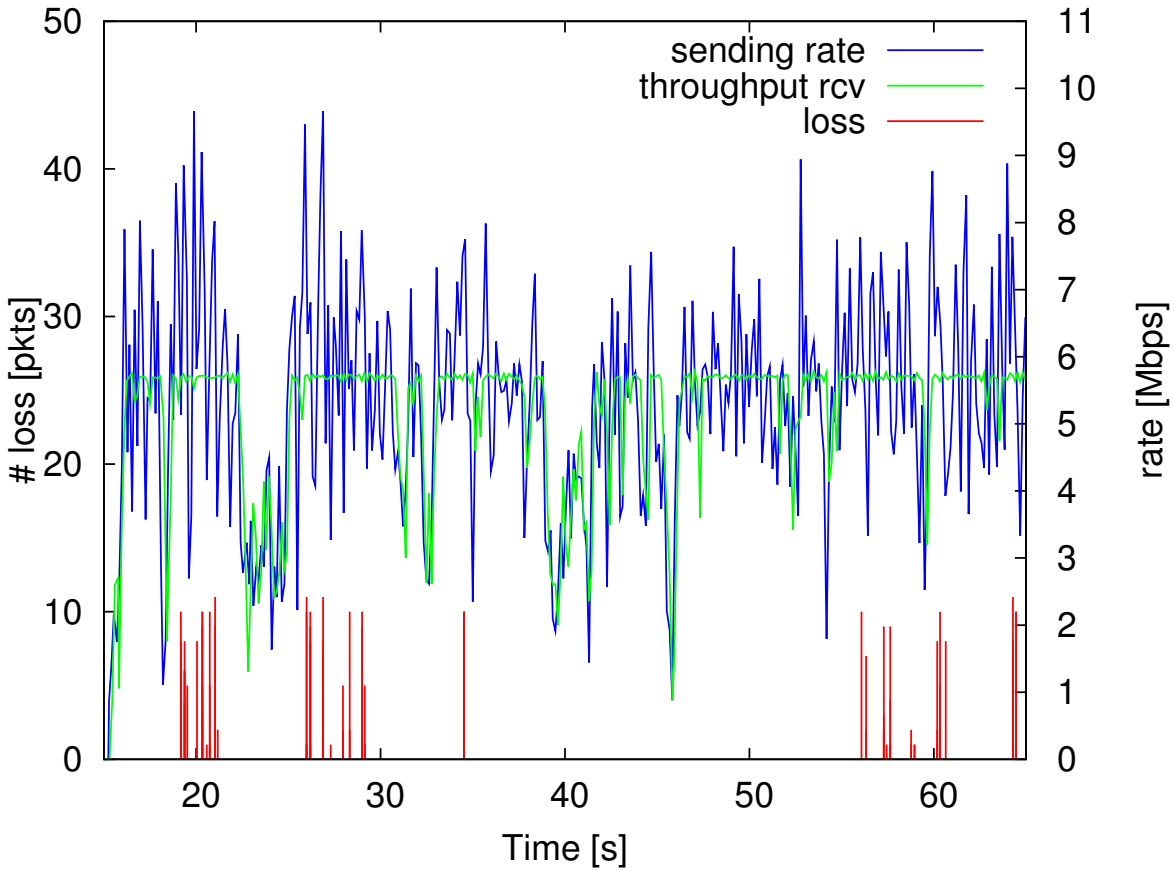
Traffic parameter: IAT = 3.2 sec, burst size = 2MB



→ Regular large burst loss at the beginning of each burst data transmission (27 packets every 3.2 seconds (=IAT))

Random Loss Pattern with Short Flows

Traffic parameter: mean IAT = 1.5 sec (NegExp), 16 - 70 packets (uniform distributed), (max.) 15 parallel flows



→ Smaller, random loss bursts depending on network load and flow size (1 - 11 packets in periods of full load)

Burst Loss Measurement

Goal

Analysis of loss pattern in real Internet usage scenarios

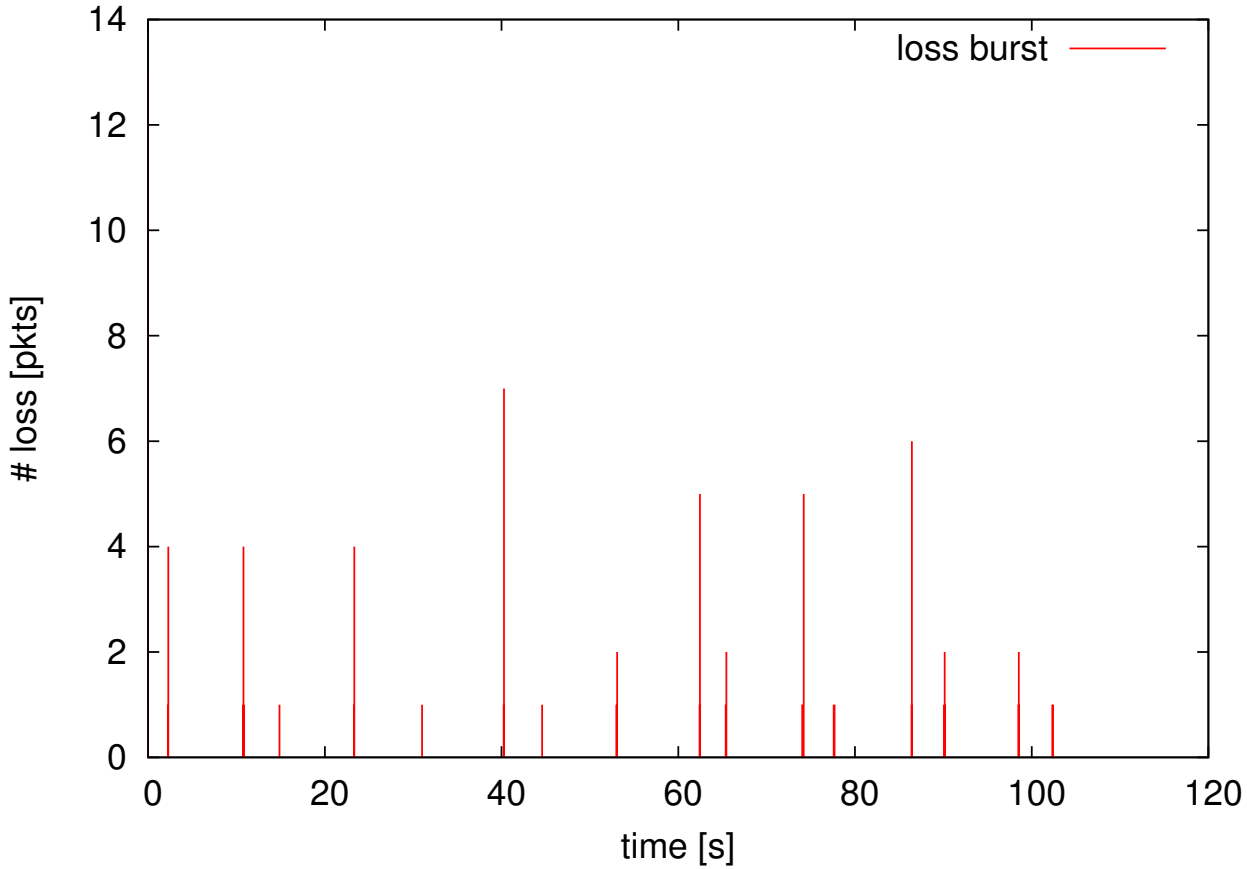
Methodology

- Traffic
  - Download: FTP download of a 80.56 MB file from a host using cubic congestion control
  - YouTube: One video with 11.59 MB
  - Web-browsing: request 33 common websites with a 12 second delay
- Measurement setup
  - Residential access network with a maximum measured datarate of 5.7 MBit/s
  - Traces of 24 trials over a single day (Sep 2012)
- Active measurement and offline evaluation

  Estimation of TCP retransmissions and burst loss metric (= number of losses within one RTT after first loss)

Burst Loss Measurement

FTP Download (one single, long flow)

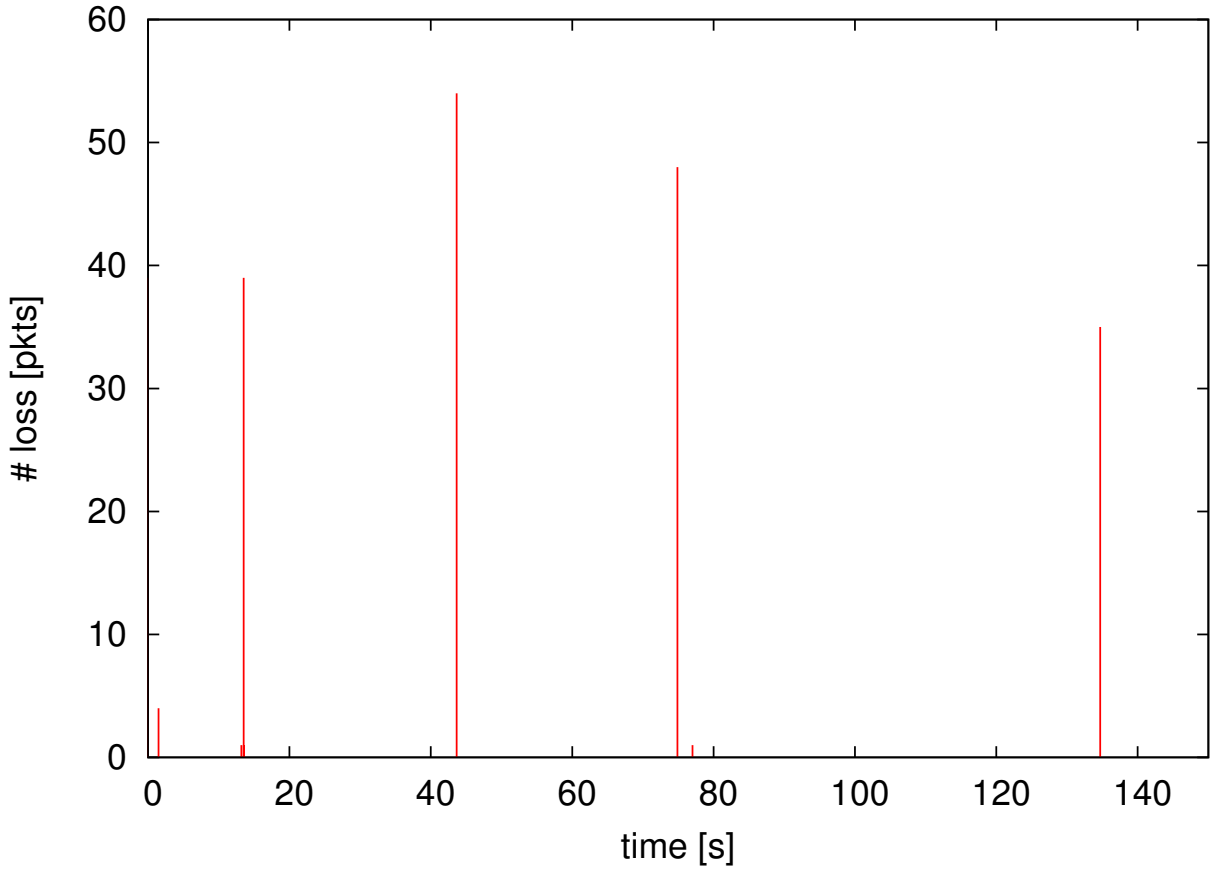


→ Quite regular burst loss pattern due to TCP cubic congestion control

Few probes showed a larger number of small burst losses (presumably caused by anomalies in the network or at the server side)

Burst Loss Measurement

YouTube (block sending)

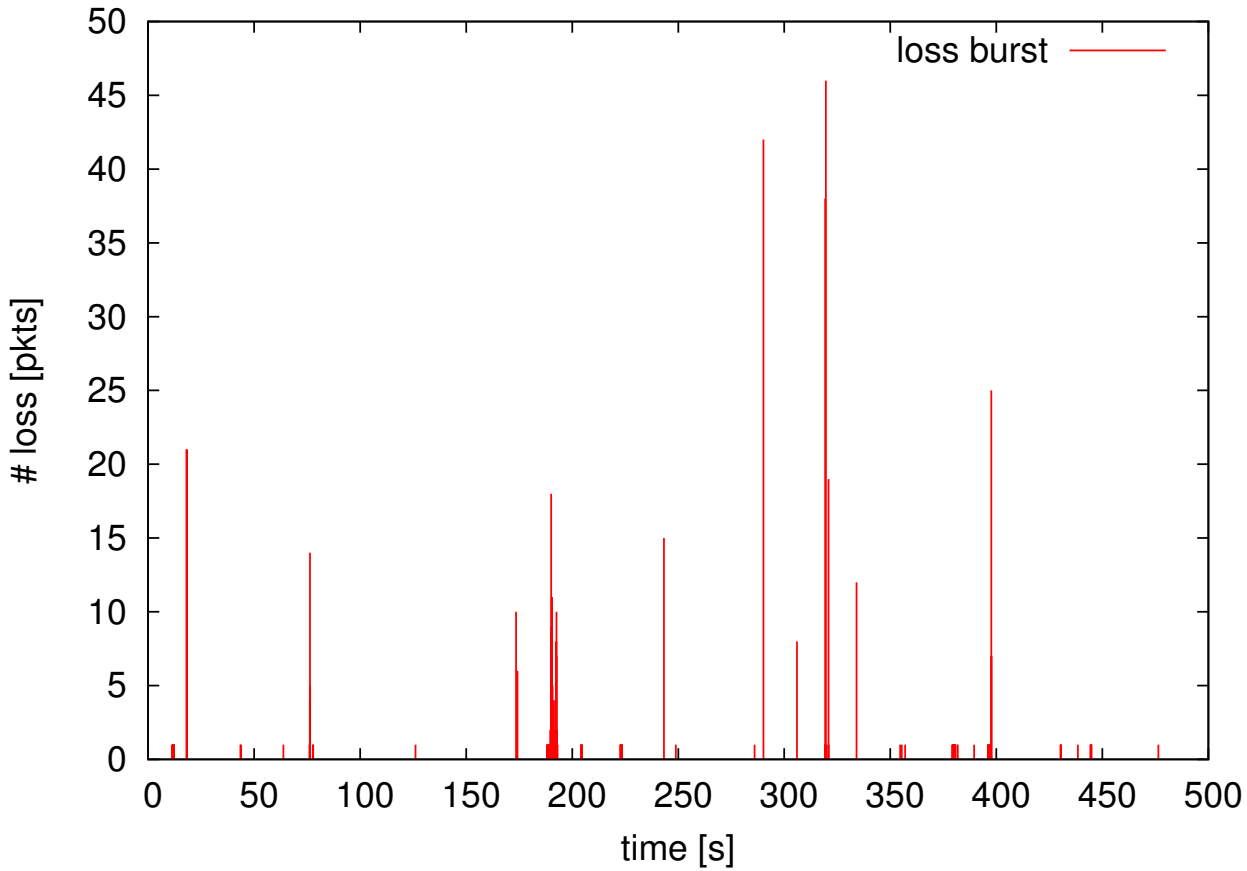


→ Fewer, but regular and larger burst losses

- 18 of 24 trials for the longer video show exactly five bursts
- Mean burst size around 33 (in both cases)

Burst Loss Measurement

Web-browsing (many short flows)



→ Irregular but few, small burst losses and very few large burst losses (as traffic model does not reflect web-browsing accurately)

Only 5.8% of flows experiencing any loss and 82.7% of bursts consist of a single loss

Conclusion

Goal

Identifying conditions and sources of loss / congestion

Approach

Loss model of transport and application behavior to differentiate losses

- caused by the host (transport and application behavior) from losses
- caused by the network or influenced by cross traffic

Model of Loss Pattern

Traffic models of greedy traffic, regular burst, and random short flows

- show very different and sometimes high loss rates (in simulation scenarios)

→ Average loss rate does not give a good indication about network state

Burst Loss Measurement

Real Internet usage scenarios (FTP download, YouTube, Web-browsing)

- show similar loss pattern

→ First step on modeling loss pattern

→ Further studies needed of more complex traffic scenarios

Outlook

- How good can a certain loss pattern be reflected using active measurements?

  How does active measurement influence the legal traffic?
- When do high loss rates influence the user Quality of Experience?

  → Loss burst metrics can provide hints on the application type